

**UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF VIRGINIA**

SONY MUSIC ENTERTAINMENT, *et al.*,

Plaintiffs,

v.

COX COMMUNICATIONS, INC., *et al.*,

Defendants.

Case No. 1:18-cv-00950-LO-JFA

**DECLARATION OF DR. NICK FEAMSTER IN SUPPORT OF COX’S REPLY
IN SUPPORT OF ITS MOTION FOR SUMMARY JUDGMENT**

I, Dr. Nick Feamster, declare as follows:

1. I am the Director of the Center for Data and Computing and Neubauer Professor in the Department of Computer Science at the University of Chicago. I submit this declaration in support of Cox’s Memorandum in Opposition to Plaintiffs’ Motion for Summary Judgment. I have personal knowledge of the facts stated in this Declaration, and if called as a witness, could testify competently to the matters contained herein.

Plaintiffs Make Several Misstatements about Hash Values

2. I have reviewed Plaintiffs’ September 24, 2019 Memorandum in Opposition to Defendants’ Motion for Summary Judgment. At numbered paragraph 12 on p.7, Plaintiffs state: “A cryptographic hash value is an alphanumeric representation of the contents of a file.” This statement is inaccurate and misleading for several reasons.

3. A hash value is a number¹ generated by performing a complex series of mathematical operations (called a “hash function”) on a given input, typically an electronic digital file.² Any particular input will always result in the same numerical output, which is called the “hash value.” It is sometimes said loosely that a hash value “represents” a particular input file, in the sense that the hash value *corresponds* to the input file. But a file’s hash value is not, in any meaningful sense, a representation of the *contents* of the file. By way of analogy, one could say that a person’s driver’s license number “represents” them in a DMV database, but no one would argue that a person’s driver’s license number was a “representation” of any of their individual characteristics.

4. Saying that a particular file has a unique hash value does not mean that a particular hash value will correspond to a unique possible input. The SHA-1, MD4, and SHA-1 Base 32 hash functions are examples of so-called “one-way functions.” This means that it is computationally infeasible to work backwards from a hash value to reconstruct the file that was used as an input to generate it. And as a technical matter, since the SHA-1, MD4, and SHA-1 Base 32 hash functions generate a hash value of a set length, but the *input* message can be arbitrarily long, there are multiple potential inputs that will yield the same hash values.³ It is thus mathematically impossible

¹ It is technically incorrect and potentially misleading to say, as Plaintiffs do, that a hash value is an “alphanumeric representation”: a hash value is simply a number. Because of how computers store and process information, however, it is common to represent numbers (especially large numbers) using base 16 or base 32 notation, not the base 10 notation that is customary in ordinary use. In base 16 and base 32 notation, letters are used as digits, and have values greater than 9 (i.e. “A” is a digit with value 10; “B” is a digit with value 11; and so on).

² A digital file can be represented numerically as a sequence of 1’s and 0’s. In essence, a hash function treats such a file as a number in order to perform mathematical operations on it.

³ For a well-designed hash function, the hash should be, for practical purposes, unique. In practice, hash values from the SHA-1, MD4, and SHA-1 Base 32 hash functions are sufficiently distinctive that they are used for error correction purposes: peer-to-peer file sharing programs can compare the hash values of the *original* file to the hash value of the *transmitted* file, to confirm that the file was not corrupted in transmission.

to determine, for a given hash value, what input was used to create it. The hash values also do not indicate a file's filetype, what the file contains, where the file originated, or whether the particular copy of the file was acquired lawfully. Simply put, it is impossible to determine anything about a file's contents from just its hash value.

**It Is Easy to Make Lawfully Acquired Files Available via Peer-to-Peer Networks,
Even Inadvertently**

5. I have reviewed the September 24, 2019 Declaration of Barbara Frederiksen-Cross, in which she describes several ways that users on peer-to-peer networks could make legally obtained files available on the network. While I generally agree with her description of the *process*, the conclusions she reaches are erroneous.

6. With respect to making files available via the Ares, eDonkey, and Gnutella networks, I agree with Ms. Frederiksen-Cross's observation that users can make files available simply by placing them in a folder that is designated for sharing. That would include files that the user obtained lawfully, such as from iTunes or from a legally purchased music CD. Depending on the particular version of the client software that is used, the user may designate one or many folders for such sharing. The client software typically comes with default settings that designate one or many folders for sharing, so that the user would not need to select a particular folder, or even know that material in the folder (or folders) is being made available on the network. Thus, it is easy to share lawfully acquired files using Ares, eDonkey, or Gnutella. Ms. Frederiksen-Cross does not dispute this.

7. I generally agree with Ms. Frederiksen-Cross's description of how a legitimately acquired file would be shared over the BitTorrent peer-to-peer network. However, Ms. Frederiksen-Cross exaggerates the difficulty of preparing a file for use with BitTorrent, and in my

view greatly underestimates the chances that two different (but identical) copies of the same file, each offering their own legally acquired copy of a work, would display the same metadata to other users of the BitTorrent network.

8. As Ms. Frederiksen-Cross acknowledges, files that are made available on the BitTorrent network may be identified by metadata that depends on a small number of parameters, including a folder name; the names of available files; and hash values for the pieces of the available files. In the case of any particular work, there is a high likelihood that all of this information will be the same, even for copies of the work that originated from different sources.

9. As an example, suppose that two different people have separately legally purchased and download the same album from iTunes, which saves the music files on their computers in a default Downloads folder. Each of them then launches the same popular BitTorrent client software, and without changing its default settings, the software creates a dot-torrent file that will enable it to make the music files available on the BitTorrent network. Because the two users' files are identical, and both were obtained from the same source (a lawful iTunes download), the names of the music files will also be identical. Because both users in this example downloaded the music files to the iTunes default Downloads folder, the folder names will almost certainly be the same. And because neither has changed the default settings on their BitTorrent client software, the software will process the music files in exactly the same way. Although Ms. Frederiksen-Cross speculates that users "might" choose different folder and file names, or "might" choose different partition sizes, in my view that is unlikely, if doing so would require the user to override the client software's default settings. In this example, if the two users were to both make their copies of the work available via BitTorrent, they would display the same metadata, even though the torrent files were created independently. And if a hypothetical third user were to unlawfully download a copy

of the file from one of those two original users, then that third user would also display the same metadata.

10. Ms. Frederiksen-Cross is incorrect when she states: “to let the world know the new dot-torrent file exists, the user must either upload it to a torrent website or provide some other way for users to obtain the dot-torrent file.” BitTorrent clients can, and do, automatically make newly created dot-torrent files available online with the click of a button. It makes sense that this would be a simple process, since otherwise BitTorrent would be much less useful, especially to ordinary users who are not technologically sophisticated.

11. As described in the prior paragraphs, it is straightforward for users of peer-to-peer networks, including BitTorrent, Ares, eDonkey and Gnutella, to share legitimately acquired content over such a network, and there is a high likelihood that multiple versions of files, created independently by different users but having identical metadata, could be available at any given time. If this were to occur, the MarkMonitor system [REDACTED] could not tell whether any particular copy of the file had been acquired lawfully or unlawfully.

12. In my view, the “term paper” analogy that Ms. Frederiksen-Cross provides to explain her views on the probability of this occurring is inaccurate and highly misleading, since it focuses on irrelevant details. Her analogy asks the reader to suppose that a teacher’s students have submitted term papers, all of which “exactly matched that of one or more other students.” She then argues that it is unlikely that the students could have independently written identical papers. But that is a completely irrelevant hypothetical. No one is arguing that peers on the BitTorrent network somehow independently created the *content* in the files. Rather, Ms. Frederiksen-Cross is arguing that if two peers have identical files, they must have obtained pieces of those files from each other. As discussed above, that is simply incorrect. Obviously, a digital copy of a digital file will have

the same “content” as the original. And nothing at all supports her speculation that every peer with a given file is likely (let alone “overwhelmingly likely”) to have copied pieces from every other peer with that file.

13. A better version of Ms. Frederiksen-Cross’s example would be if a teacher instructed her students to transcribe one of her lectures, then discovered that all of the transcripts had the same content in the same *format*, with the same font size, typeface, and margins. Similarities such as these would be completely unsurprising, given that the most popular word-processing programs generally ship with identical default settings, which there is seldom any reason to change. Nothing about such similarities in formatting would indicate that the students had copied transcripts from each other.

14. It is well known that lawfully acquired files may be shared inadvertently over peer-to-peer networks. For example, in 2007, the House of Representatives Committee on House Oversight and Government Reform held hearings after it was discovered that sensitive military secrets had inadvertently been shared by military personnel using the LimeWire file sharing program.⁴ LimeWire uses the Gnutella network, as well as the BitTorrent protocol.

15. Ms. Frederiksen-Cross also argues that [REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED] Not surprisingly, she offers no evidence or analysis to support that view, which simply makes no sense for a number of reasons.

⁴ See House Oversight & Govt. Reform, *Inadvertent File Sharing*, CQ Congressional Testimony (July 24, 2007) (Statement of Robert Boback); Hiawatha Bray, “File-Sharing Imperils US Secrets: Use of the Software in the Military May Expose Documents,” *Boston Globe* (Aug. 5, 2004).

16. Primarily, Ms. Frederiksen-Cross points to no evidence that any of the peers that MarkMonitor supposedly detected were in contact *with each other*. In Paragraph 10 of her declaration, she refers misleadingly to “users on a peer-to-peer file-sharing network,” which makes it sound like there is (for example) a single unified BitTorrent network, to which all BitTorrent users are connected and on which they all share files simultaneously with every other user. That is not the way it works: there is no unitary “BitTorrent network” (or other “peer-to-peer network”) in that sense. Terms such as “BitTorrent network” (or the like) refer generally to file sharing that uses a particular peer-to-peer *protocol* through which users are *able* to (but not *required* to) connect with certain other users. The BitTorrent protocol, for example, enables users of BitTorrent software to form temporary network connections with a constantly shifting subset of other BitTorrent users (or peers) that wish to exchange a given file.⁵ As I explain in my Report, these shifting groups of networked peers are called “swarms.” (Feamster Rpt. ¶ 62.) Swarms are formed “on the fly,” and different swarms can form to share different (but identical) copies of the same file. A number of such swarms may be active at the same time. In addition, because connections between peers are constantly being made and broken, the structure of a swarm shifts and changes over time.

17. Thus, the mere fact that two BitTorrent users appear to have the same file available at the same time does not indicate that the users are part of the same “network” in the sense that Ms. Frederiksen-Cross appears to be implying in her September 24, 2019 Declaration.

18. Logically, the more popular a file is, the more likely that there will be multiple swarms sharing it.

⁵ I explain this process in more detail in the May 15, 2019 Rebuttal Expert Report of Dr. Nick Feamster, Ph.D. (“Feamster Rpt.”), at ¶¶ 60-87.

19. In addition, the fact that popular files are widely available would logically make it *more* likely that at least some of them originated from different sources, not less likely.

20. Ms. Frederiksen-Cross's statement that separate observations somehow show users acting "simultaneously," even though the observations were conducted at different times over the course of an entire month, is clearly incorrect.

Plaintiffs Misconstrue the "Tit for Tat" Incentive Model

21. Plaintiffs are incorrect that just because a peer can be located on a BitTorrent network, that means that the peer is actively downloading and uploading content, or that it doing either of those things. In fact, as I have discussed elsewhere (and as Ms. Frederiksen-Cross has repeatedly admitted), the MarkMonitor system is an example of a BitTorrent peer that routinely does not download content (at least as configured for the version used for the RIAA that is relevant to this lawsuit). Nor does it upload any content.

22. Plaintiffs misinterpret my deposition testimony discussing the so-called "tit for tat" model that many BitTorrent clients employ to create incentives. Plaintiffs rely on an out-of-context statement from my deposition that "[i]n the BitTorrent protocol, any participating peer will **generally** be downloading or retrieving pieces of the file as well as ... providing other pieces of the file to other pieces in the swarm." I used the term "generally" advisedly.

23. The "tit for tat" incentive model is just that: a model. It is employed to counteract a problem common to peer-to-peer networks, which is that once a peer succeeds in retrieving the file or other content that it wants, it may simply leave the network. To quote the designer of BitTorrent, Bram Cohen, "Frequently downloaders cease uploading as soon as their download completes, although it is considered polite to leave the client uploading for awhile after your

download completes.”⁶ As I discuss in my Report, clients often leave the network immediately after they have retrieved a file of interest. Feamster Rpt., at ¶ 76. This creates a problem for BitTorrent because when a peer leaves the network, its copy of the content is unavailable to other peers. A similar problem occurs if a peer only downloads pieces of a file, without making them available for others to upload. The so-called “tit for tat” model is designed to provide incentives to a peer to make pieces of files available for others. The model does not, however, *require* any other peer to actually download pieces that the first peer makes available. Nor does it *require* the first peer to actually upload pieces from any other peer.

24. One way to confirm that a peer is actually downloading or uploading pieces of a file would be to engage with the peer on a BitTorrent network, and attempt to actually download content from them. The MarkMonitor system [REDACTED]

[REDACTED].

25. Before one peer can obtain data from, or transmit data to, another peer, they must perform a “handshake,” exchange certain information (but no file content), and decide whether to initiate an exchange of file data. For a variety of reasons, it is possible that no such exchange will occur. There are many reasons that a peer might not transmit any pieces of a file that it appears to have available (or partially available). Similarly, there are reasons that a peer which *lacks* pieces of a file might not download pieces from other peers in the swarm. Among other things:

- Pieces of a file can be obtained from *any* peer in the swarm that has them. Thus, just because one particular peer happens to have a piece available, that does not mean any other peer will request (or obtain) the piece from that peer.
- If other peers already have a given piece, they will not request or download the piece again. Thus, merely because one particular peer happens to have one particular piece of a

⁶ Braham Cohen, Incentives build robustness in BitTorrent, WORKSHOP ON ECONOMICS OF PEER-TO-PEER SYSTEMS (Vol. 6, pp. 68-72) (June 2003).

file available, that does not mean any other peer needs it. Such a piece would not be transmitted.

- It is incorrect to say, as Ms. Frederiksen-Cross does, that a peer-to-peer client “automatically distributes” content. A peer will not transmit a piece of a file unless another peer requests it.
- For very popular files, there are likely to be many peers who have pieces of the file, decreasing the likelihood that any given peer will be asked for a piece.
- As I explained in my Report (see ¶¶ 75-81), i [REDACTED]

26. There are additional situations where a “tit for tat” incentive model is not in play. For example, a peer that already has 100% of a file will not seek to download pieces that it already has. In addition, merely because that peer may be making all the pieces of the file *available*, the “tit for tat” model would not require any other peer to actually download any of the pieces from it.⁷

27. Even if BitTorrent clients do download or upload content, contrary to what Plaintiffs argue, they do not do so “simultaneously.” Plaintiffs appear to have based this on a misreading of a paragraph taken out of context from my Report, in which I stated:

[REDACTED]

[REDACTED] Again, this does not mean that BitTorrent clients are continuously and simultaneously either transmitting or receiving data from other peers. Nor does it say (or imply) that a BitTorrent client will necessarily do either of those things. The context of the statement—

⁷ Similarly, it is worth noting that a peer that does not have *any* pieces of a file obviously cannot make any pieces available, and thus cannot “swap” pieces—nor could it *ever* “swap” pieces with the first peer it encounters, since that peer would of necessity already have the same pieces. It is thus evident that a “tit for tat” exchange cannot simply be rigidly enforced.

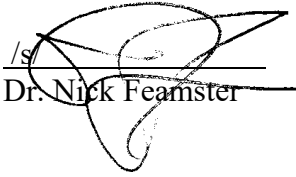
which describes, in general terms, the process of an BitTorrent file exchange, and explains the model that BitTorrent uses to create *incentives* for sharing—makes that even clearer.

28. Merely because a peer has pieces of a file available, that does not mean that the pieces contain any pieces of a copyrighted work. Depending on the file and how it has been prepared for use on the BitTorrent network, many of the pieces are likely to consist entirely or partly of data such as hash values of files or file pieces, or other metadata. In addition, as I explained in my Report at ¶¶ 75-81, where a peer does not have the complete file, it is most likely the missing pieces are not contiguous but rather spread throughout the file—making it impossible in many cases for the user to decode and play even the partial file.

29. For all of these reasons, even if the MarkMonitor system observes a user claiming to have a file available, on a device active at a Cox subscriber's IP address, that merely indicates that the file (assuming it actually exists) is being made available for sharing. It does not provide any evidence whatsoever that the file, or any part of it, is (or was) actually being downloaded by anyone in the peer-to-peer network (including the peer who is being observed). Nor does it provide any evidence to show whether the file was obtained lawfully or unlawfully, or from what source.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

Executed this 11th day of October, 2019 in Chicago, Illinois.


/s/
Dr. Nick Feamster